Convergence Analysis of a Parallel Newton Scheme for Dynamic Power Grid Simulations

Brett A. Robbins, *Student Member, IEEE*, Victor M. Zavala, *Member, IEEE*

***Abstract*—We analyze the convergence properties of a parallel Newton scheme for differential systems. The scheme concurrently solves the time-coupled nonlinear systems arising from the application of implicit discretization schemes. We have found that the scheme acts as a tracking algorithm that converges to a moving manifold given by the solution of the nonlinear system at the current time step parameterized in the iterating solution of the previous step. This property explains why the method can significantly reduce the number of iterations compared with the sequential Newton method that marches forward in time. The method exhibits a theoretical lower bound on the number of iterations equal to the number of discretization points. A numerical study using a detailed dynamic power grid model is provided to demonstrate the scalability of the method.**

***Index Terms*—Parallel Processing, Dynamic Simulation, Power Systems, Newton's Method, Convergence**

I. Introduction

THE U.S. power grid is expected to sustain highly volatile environments as a result of large adoptions of intermittent renewable power and price-responsive demands. Understanding of transient phenomena arising under these environments has been hindered by the computational complexity of the associated dynamic models. For instance, in the 2003 blackout report produced by the U.S. Department of Energy, limited computational resources prevented a more detailed analysis of the dynamic phenomena that triggered cascading events [1]. This situation indicates that, although high-performance computing systems have evolved significantly during recent years, they have not been fully exploited to address existing needs in power grid simulation. Motivated by these limitations, we analyze the convergence properties of a parallel method for large-scale dynamic simulation.

Parallel methods for dynamic power grid simulation date back to [2] and can be broadly classified as parallel-in-time, parallel-in-space, and combinations of both. An extensive review can be found in [3].

The majority of the proposed methods can be classified as parallel in space strategies, where weak coupling between buses is exploited to partition the network [4]–[6]. Alternatively, the approaches in [7]–[10] decouple the quasi-steady-state network equations from the differential equations.

Parallel-in-time simulation strategies have been reported in [11]–[14]. Similar to space partitioning, these relaxation (e.g., Gauss-Jacobi) methods allow the time horizon to be partitioned and solved concurrently by dropping the coupling terms in the Newton system between neighboring times steps. Such methods are also referred to as parallel block Newton or multi-splitting methods [15]–[17]. Empirically, it has been observed that parallel in time Newton methods can significantly reduce the number of iterations compared with sequential Newton methods. In addition, parallel methods can help overcome memory bottlenecks associated to full-space Newton methods. An existing limitation, however, is that the convergence properties of parallel block Newton methods cannot be easily analyzed with traditional techniques.

This work provides insights on the convergence properties of a parallel in time block Newton method and on efficiency gains over the sequential method. We have found that the parallel Newton scheme acts as a tracking algorithm that converges rapidly to a moving manifold given by the solution of the nonlinear system of current time step parameterized by the iterating solution at the previous step. This property explains why the method can significantly reduce the number of iterations compared with sequential methods.

The paper is structured as follows. Sections II and III describe the dynamic simulation setting. Section IV provides convergence results for the parallel Newton method. A differential algebraic model for power systems is presented in Section V. This model is used to demonstrate the performance of the parallel method in Section VI. Section VII concludes this paper and provides directions for future work.

II. Setting

We consider the differential-algebraic (DAE) system

$$\dot{z}(t) = f(z(t), y(t)), \qquad z(0) = z_0 \tag{2a}$$
$$0 = g(z(t), y(t)), \qquad t \in [0, T], \tag{2b}$$

where $z(\cdot) \in \mathbb{R}^{n_z}$ are the differential states with initial conditions z_0, $y(\cdot) \in \mathbb{R}^{n_y}$ are the algebraic states, t is the scalar time dimension, and T is the final time. The mappings $f : \mathbb{R}^{n_z} \times \mathbb{R}^{n_y} \to \mathbb{R}^{n_z}$ and $g : \mathbb{R}^{n_z} \times \mathbb{R}^{n_y} \to \mathbb{R}^{n_y}$ are assumed to be at least twice continuously differentiable. The DAE system is assumed to be index one.

We seek to solve this system using an implicit discretization approach such as implicit Euler, Gauss, and Radau collocation [18]. In the case of implicit Euler, for instance, we obtain a set

V. M. Zavala is with the Mathematics and Computer Science Division at Argonne National Laboratory, Argonne, IL 60439. e-mail: zavala@mcs.anl.gov

B. A. Robbins is with the Department of Electrical Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, 61801. e-mail: robbins3@illinois.edu.

$$\mathbf{\Phi}'(x) = \begin{bmatrix} \nabla_{x_1}\Phi(x_1,x_0) & & & & \\ \nabla_{x_1}\Phi(x_2,x_1) & \nabla_{x_2}\Phi(x_2,x_1) & & & \\ & & \ddots & & \\ & & & \nabla_{x_{N-1}}\Phi(x_{N-1},x_{N-2}) & \\ & & & \nabla_{x_{N-1}}\Phi(x_N,x_{N-1}) & \nabla_{x_N}\Phi(x_N,x_{N-1}) \end{bmatrix} \tag{1}$$

of nonlinear equations of the following form:

$$\begin{aligned} \Phi_z(z_{k+1},y_{k+1},z_k) &= z_{k+1} - z_k - h_k \cdot f(z_{k+1},y_{k+1}) \\ &= 0, \end{aligned} \tag{3a}$$

$$\begin{aligned} \Phi_y(z_{k+1},y_{k+1},z_k) &= g(z_{k+1},y_{k+1}) \\ &= 0, \end{aligned} \tag{3b}$$

for $k = 0, \ldots, N-1$. Here, N is the number of discretization steps of length h_k, and we have $\sum_{k=0}^{N-1} h_k = T$. If we group the differential and algebraic states into a single vector $x_k \in \mathbb{R}^{n_z+n_y}$, $k = 0, ..., N$ we can write the system in the following general form:

$$\Phi(x_{k+1}, x_k) = 0, \quad k = 0, ..., N-1. \tag{4}$$

We will refer to the above system as the *full space system*. Its solution is be denoted as x_k^*, $k = 0, ..., N$. The Jacobian of the mapping $\Phi : \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \to \mathbb{R}^{n_x}$ with respect to the first argument (x_{k+1}) is denoted as $\Phi' := \nabla_{x_{k+1}}\Phi : \mathbb{R}^{n_x} \times \mathbb{R}^{n_x} \to \mathbb{R}^{n_x \times n_x}$. If an implicit Euler discretization scheme is used, the Jacobian terms of the full-space system have the following structure,

$$\begin{aligned} &\Phi_z'(z_{k+1},y_{k+1},z_k) \\ &\quad = \mathbb{I}_{n_z} - h_k(\nabla_z f(z_{k+1},y_{k+1}) + \nabla_y f(z_{k+1},y_{k+1})) \end{aligned} \tag{5a}$$

$$\begin{aligned} &\Phi_y'(z_{k+1},y_{k+1},z_k) \\ &\quad = \nabla_z g(z_{k+1},y_{k+1}) + \nabla_y g(z_{k+1},y_{k+1}) \end{aligned} \tag{5b}$$

for $k = 0, \ldots, N-1$.

III. Parallel Newton Method

To solve the full-space system (4), we use a *parallel-in-time Newton scheme*. The iteration j takes the form

$$\Phi'(x_{k+1}^j, x_k^j)\Delta x_{k+1}^j = -\Phi(x_{k+1}^j, x_k^j), \; k = 0, ..., N-1, \tag{6}$$

where

$$x_{k+1}^{j+1} = x_{k+1}^j + \Delta x_{k+1}^j, \; k = 0, ..., N-1. \tag{7}$$

Here, x_0 is the fixed initial condition. The stopping criterion is the convergence of the *global* error,

$$\left\| \begin{bmatrix} \Phi(x_1^j, x_0^j) \\ \vdots \\ \Phi(x_N^j, x_{N-1}^j) \end{bmatrix} \right\|_\infty \leq \epsilon. \tag{8}$$

We note that the Newton step for all time steps $k = 0, ..., N-1$ can be computed in parallel. The method can be interpreted as a Newton scheme applied to the full-space system (4) that drops the time coupling terms from the Jacobian matrix. To see this, we define the full-space vector $x^T = [x_1^T, ..., x_N^T]^T$ and system,

$$\mathbf{\Phi}(x) = \begin{bmatrix} \Phi(x_1, x_0) \\ \Phi(x_2, x_1) \\ \vdots \\ \Phi(x_N, x_{N-1}) \end{bmatrix}. \tag{9}$$

The Jacobian for the full-space system is given in equation (1). The full-space Newton iteration is $x^{j+1} = x^j + \mathbf{\Phi}'(x^j)^{-1}\mathbf{\Phi}(x^j)$. By dropping the off-diagonal terms from the Jacobian and by noticing that $\Phi'(x_{k+1}, x_k) = \nabla_{x_{k+1}}\Phi(x_{k+1}, x_k)$ we recover the parallel Newton scheme (6). The advantage of the full-space Newton method it that it gives a fast convergence rate. A key limitation, however, is the increasing computational times and memory requirements as the number of discretization points is extended.

We also highlight the difference of the parallel Newton scheme from the *sequential* Newton scheme that marches forward in time. This scheme solves the system $\Phi(x_1, x_0, h_0)$ starting at $k = 0$ to obtain x_1^* and then solves $\Phi(x_2, x_1^*, h_1)$ to obtain x_2^* and so on. In other words, the system iterates on

$$\Phi'(x_{k+1}^j, x_k^*)\Delta x_{k+1}^j = -\Phi(x_{k+1}^j, x_k^*), \; k = 0, ..., N-1. \tag{10}$$

The stopping criterion of this method is based on the convergence of the *local* errors,

$$\left\| \; \Phi(x_{k+1}^j, x_k^*) \; \right\|_\infty \leq \epsilon, \quad k = 0, ..., N-1. \tag{11}$$

One of the advantages of this approach is that it is not as memory demanding as a full-space Newton scheme. A limitation of the sequential scheme, however, is that it wastes computational time by tightening the error of the *local* nonlinear system at time step k when this might not be necessary for the next system $k+1$. In other words, the scheme lacks a *global* view. This is particularly inefficient in simulation-based optimization where low-precision simulations are often needed. The parallel Newton scheme iterates simultaneously over the entire set of nonlinear equations and monitors the global error (8) as in the full-space Newton method. Consequently, significant amounts of computational time can be saved. In addition, the scheme can be run in a distributed memory system and thus can accommodate long time horizons and fine discretization resolutions.

IV. Convergence Analysis

A key observation that we make in this paper is that the parallel Newton scheme can be interpreted as a Newton scheme under *parametric perturbations*. As we will show, this interpretation is key in assessing the convergence properties and computational limitations of the scheme.

To start the discussion, we consider the local system at time step $k+1$,

$$\Phi(x_{k+1}, x_k) = 0. \tag{12}$$

The Newton scheme tries to find x_{k+1}^* by linearizing this system around the current iterate x_{k+1}^j and treats x_k as an *exogenous* parameter with current value x_k^j. The Newton system is

$$\Phi(x_{k+1}^j, x_k^j) + \Phi'(x_{k+1}^j, x_k^j)(x_{k+1}^{j+1} - x_{k+1}^j) = 0. \tag{13}$$

Here, x_k^j is the exogenous parameter sequence over the iteration sequence $j = 0, ..., J_k$ converging to a limit point x_k^*. For $k = 0$ we have $x_0^j = x_0$, which are the initial conditions.

Consider the following *perturbed* system,

$$\Phi(x_{k+1}, x_k^*) = r, \tag{14}$$

where r is a residual or perturbation. The solution of this system is denoted as $x_{k+1}(r)$ an satisfies $x_{k+1}^* = x_{k+1}^*(x_k^*) = x_{k+1}(0)$.

Definition 1: (**Strong Regularity.**) The system (14) is said to be strongly regular at x_{k+1}^* [19] if there exists $L > 0$ such that

$$\|x_{k+1}(r) - x_{k+1}^*\| \le L\|r\|. \tag{15}$$

In [19], it is shown that a condition for the system (14) to be strongly regular is that the derivative matrix $\Phi'(x_{k+1}^*, x_k^*)$ is nonsingular.

The Newton system (13) at iteration j can be posed in the form of (14) by adding and substracting $\Phi(x_{k+1}^{j+1}, x_k^*)$,

$$\begin{aligned}\Phi(x_{k+1}^{j+1}, x_k^*) = \Phi(x_{k+1}^{j+1}, x_k^*) - \Phi(x_{k+1}^j, x_k^j) \\ - \Phi'(x_{k+1}^j, x_k^j)(x_{k+1}^{j+1} - x_{k+1}^j).\end{aligned} \tag{16}$$

Under strong regularity we have that

$$\|x_{k+1}^{j+1} - x_{k+1}^*\| \le L\|r_{k+1}^j\|, \tag{17}$$

where

$$\begin{aligned}r_{k+1}^j = \Phi(x_{k+1}^{j+1}, x_k^*) - \Phi(x_{k+1}^j, x_k^j) \\ - \Phi'(x_{k+1}^j, x_k^j)(x_{k+1}^{j+1} - x_{k+1}^j).\end{aligned} \tag{18}$$

Theorem 1: (**Convergence of Perturbed Newton System**.) Assume that the system (14) is strongly regular at x_{k+1}^*, x_k^*. Assume also that the exogenous sequence x_k^j converges to the limit point x_k^*. If there exists $\sigma > 0$ such that $\|x_k^j - x_k^*\| \le \sigma\|x_{k+1}^{j+1} - x_{k+1}^j\|$, then the Newton scheme converges superlinearly to x_{k+1}^*. If $\|x_k^j - x_k^*\| \le \sigma\|x_{k+1}^{j+1} - x_{k+1}^j\|^2$ and the mapping Φ' is Lipschitz in both arguments, then convergence is quadratic.

Proof: We first note that the residual of the Newton system can be expanded as

$$\begin{aligned}&r_{k+1}^j \\ &= \Phi(x_{k+1}^{j+1}, x_k^*) - \Phi(x_{k+1}^j, x_k^j) - \Phi'(x_{k+1}^j, x_k^j)(x_{k+1}^{j+1} - x_{k+1}^j) \\ &= \Phi(x_{k+1}^{j+1}, x_k^*) - \Phi(x_{k+1}^j, x_k^*) - \Phi'(x_{k+1}^j, x_k^*)(x_{k+1}^{j+1} - x_{k+1}^j) \\ &+ \Phi(x_{k+1}^{j+1}, x_k^j) - \Phi(x_{k+1}^j, x_k^j) - \Phi'(x_{k+1}^j, x_k^j)(x_{k+1}^{j+1} - x_{k+1}^j) \\ &- \Phi(x_{k+1}^{j+1}, x_k^j) + \Phi(x_{k+1}^j, x_k^*) + \Phi'(x_{k+1}^j, x_k^*)(x_{k+1}^{j+1} - x_{k+1}^j) \\ &+ \Phi(x_{k+1}^{j+1}, x_k^*) - \Phi(x_{k+1}^{j+1}, x_k^*).\end{aligned} \tag{19}$$

We define the exogenous perturbation term,

$$\begin{aligned}&w_{k+1}^j = \\ &\quad \Phi(x_{k+1}^{j+1}, x_k^j) - \Phi(x_{k+1}^j, x_k^j) - \Phi'(x_{k+1}^j, x_k^j)(x_{k+1}^{j+1} - x_{k+1}^j) \\ &+ \Phi(x_{k+1}^j, x_k^*) - \Phi(x_{k+1}^{j+1}, x_k^*) + \Phi'(x_{k+1}^j, x_k^*)(x_{k+1}^{j+1} - x_{k+1}^j) \\ &+ \Phi(x_{k+1}^{j+1}, x_k^*) - \Phi(x_{k+1}^{j+1}, x_k^j),\end{aligned} \tag{20}$$

so that

$$\begin{aligned}r_{k+1}^j = \Phi(x_{k+1}^{j+1}, x_k^*) - \Phi(x_{k+1}^j, x_k^*) \\ - \Phi'(x_{k+1}^j, x_k^*)(x_{k+1}^{j+1} - x_{k+1}^j) + w_{k+1}^j.\end{aligned} \tag{21}$$

Applying the mean value theorem and bounding, we have

$$\begin{aligned}&\|r_{k+1}^j\| \\ &\le \sup_{\gamma\in[0,1]} \{\Phi'(\gamma x_{k+1}^{j+1} + (1-\gamma)x_{k+1}^j, x_k^*) - \Phi'(x_{k+1}^j, x_k^*)\}\cdot \\ &\qquad \cdot \|\Delta x_{k+1}^j\| + \|w_{k+1}^j\| \\ &= \kappa_1\|\Delta x_{k+1}^j\| + \|w_{k+1}^j\|.\end{aligned} \tag{22}$$

The exogenous term is bounded as,

$$\begin{aligned}&\|w_{k+1}^j\| \\ &\le \sup_{\gamma\in[0,1]} \{\Phi'(\gamma x_{k+1}^{j+1} + (1-\gamma)x_{k+1}^j, x_k^j) - \Phi'(x_{k+1}^j, x_k^j)\}\cdot \\ &\qquad \cdot \|\Delta x_{k+1}^j\| \\ &+ \sup_{\gamma\in[0,1]} \{\Phi'(x_{k+1}^j, x_k^*) - \Phi'(\gamma x_{k+1}^{j+1} + (1-\gamma)x_{k+1}^j, x_k^*)\}\cdot \\ &\qquad \cdot \|\Delta x_{k+1}^j\| \\ &+ L_\Phi\|x_k^j - x_k^*\| \end{aligned} \tag{23}$$

$$\le \kappa_2\|\Delta x_{k+1}^j\| + \kappa_3\|x_k^j - x_k^*\|, \tag{24}$$

so that

$$\|r_{k+1}^j\| \le (\kappa_1 + \kappa_2)\|\Delta x_{k+1}^j\| + \kappa_3\|x_k^j - x_k^*\|. \tag{25}$$

We first establish *superlinear* convergence from

$$\begin{aligned}&\|x_{k+1}^{j+1} - x_{k+1}^*\| \\ &\le L\|r_{k+1}^j\| \\ &\le L(\kappa_1 + \kappa_2)\|\Delta x_{k+1}^j\| + L\kappa_3\|x_k^j - x_k^*\| \\ &= L(\kappa_1 + \kappa_2)\|x_{k+1}^{j+1} - x_{k+1}^j\| + L\kappa_3\|x_k^j - x_k^*\| \\ &= L(\kappa_1 + \kappa_2)\|x_{k+1}^{j+1} - x_{k+1}^* + x_{k+1}^* - x_{k+1}^j\| \\ &\qquad + L\kappa_3\|x_k^j - x_k^*\|.\end{aligned} \tag{26}$$

From the assumptions we have

$$\begin{aligned}\|x_k^j - x_k^*\| &\leq \sigma \|\Delta x_{k+1}^j\| \\ &\leq \sigma \left(\|x_{k+1}^{j+1} - x_{k+1}^*\| + \|x_{k+1}^j - x_{k+1}^*\| \right) \end{aligned} \quad (27)$$

and

$$\begin{aligned}\|x_{k+1}^{j+1} - x_{k+1}^*\| &\leq L(\kappa_1 + \kappa_2 + \sigma\kappa_3) \cdot \\ &\cdot \left(\|x_{k+1}^{j+1} - x_{k+1}^*\| + \|x_{k+1}^j - x_{k+1}^*\| \right). \end{aligned} \quad (28)$$

Finally,

$$\|x_{k+1}^{j+1} - x_{k+1}^*\| \leq \frac{\alpha}{1-\alpha} \|x_{k+1}^j - x_{k+1}^*\|. \quad (29)$$

Here, $\alpha = L(\kappa_1 + \kappa_2 + \sigma\kappa_3)$. Consequently, the sequence converges superlinearly.

To establish quadratic convergence, we notice that if the derivative mapping is Lipschitz continuous, then

$$\|r_{k+1}^j\| \leq \kappa_1 \|\Delta x_{k+1}^j\|^2 + \|w_{k+1}^j\|. \quad (30)$$

Moreover, we have that

$$\|w_{k+1}^j\| \leq \kappa_2 \|\Delta x_{k+1}^j\|^2 + \kappa_3 \|x_k^j - x_k^*\|. \quad (31)$$

From the assumptions we have that $\|x_k^j - x_k^*\| \leq \sigma \|x_{k+1}^{j+1} - x_{k+1}^j\|^2$. Consequently,

$$\begin{aligned} &\|x_{k+1}^{j+1} - x_{k+1}^*\| \\ &\leq L(\kappa_1 + \kappa_2) \|x_{k+1}^{j+1} - x_{k+1}^* + x_{k+1}^* - x_{k+1}^j\|^2 \\ &+ L\kappa_3 \|x_k^j - x_k^*\|^2. \end{aligned} \quad (32)$$

Expanding the squared term and dividing through by $\|x_{k+1}^{j+1} - x_{k+1}^*\|$, we obtain

$$\begin{aligned} &\frac{1}{\|x_{k+1}^{j+1} - x_{k+1}^*\| + 2 \cdot \|x_{k+1}^j - x_{k+1}^*\| + \frac{\|x_{k+1}^j - x_{k+1}^*\|^2}{\|x_{k+1}^{j+1} - x_{k+1}^*\|}} \\ &\leq L(\kappa_1 + \kappa_2 + \sigma\kappa_3). \end{aligned} \quad (33)$$

The sequence on the left-hand side is bounded only if there exists $\gamma > 0$ such that

$$\frac{\|x_{k+1}^j - x_{k+1}^*\|^2}{\|x_{k+1}^{j+1} - x_{k+1}^*\|} \geq \gamma, \quad (34)$$

which implies

$$\|x_{k+1}^{j+1} - x_{k+1}^*\| \leq \gamma \cdot \|x_{k+1}^j - x_{k+1}^*\|^2. \quad (35)$$

The proof is complete □.

We note that once $x_k^j = x_k^*$, then $\|w_k^j\| = 0$ and $\alpha = L\kappa_1$ so that the pure Newton convergence rate is recovered. The same holds for the sequential Newton scheme. The above result looks into the *local* equation system of a time step k and states that the Newton iteration *absorbs* the parametric perturbations induced by the incoming data x_k^j and converges to the local solution x_{k+1}^*. As can be seen, the convergence rate is dictated by the error of the previous time step. We now establish the convergence of the parallel Newton scheme for the full system of nonlinear equations (4).

Theorem 2: (**Convergence of Full System**). Assume that the conditions of Theorem 1 hold for $k = 0, ..., N-1$. Define the local errors $\epsilon_k^j := \|x_k^j - x_k^*\|,\ k = 0, ..., N$ with initial values $\epsilon_k^0 > 0,\ k = 0, ..., N$. The iterations of the parallel Newton scheme (6) converge to the solution of the system (4) $x_k^*,\ k = 0, ..., N$. Furthermore, the minimum number of iterations is N.

Proof: We have the contraction condition

$$\|x_{k+1}^{j+1} - x_{k+1}^*\| \leq \beta(\|x_{k+1}^j - x_{k+1}^*\|) + \omega \|x_k^j - x_k^*\| \quad (36)$$

for $k = 0, ..., N-1$, where $\epsilon_0^j = 0$ since x_0 is fixed. Thus, we have the following iteration sequence:

$$\begin{aligned} \epsilon_1^{j+1} &\leq \beta \epsilon_1^j \\ \epsilon_2^{j+1} &\leq \beta \epsilon_2^j + \omega \epsilon_1^j \\ &= \beta \epsilon_2^j + \omega \beta \epsilon_1^{j-1} \\ &\vdots \\ \epsilon_N^{j+1} &\leq \beta \epsilon_N^j + \omega \epsilon_{N-1}^j \\ &= \beta \epsilon_N^j + \beta \sum_{i=1}^{N-1} \omega^i \epsilon_{N-i}^{j-i}. \end{aligned} \quad (37)$$

We know that $\epsilon_1^j \to 0$. Applying a recursive argument, we obtain $\epsilon_k^j \to 0,\ k = 0, ..., N$, so that the scheme converges to $x_k^*,\ k = 0, ..., N$. From the above sequence we see that the local error for $k = N$ and iteration j depends on the *delayed* error at $k = N-1$, which in turn depends on the delayed error at $k = N-2$ and so on. Since the error at each time step k takes at least one iteration to converge once the error at $k-1$ converges, the perturbation term $\sum_{i=1}^{N-1} \omega^i \epsilon_{N-i}^{j-i}$ requires at least $N-1$ iterations to converge. Consequently, the local error e_N^j takes at least N iterations to converge, and so does the global error. □

The previous convergence results shed some light on the convergence properties of the parallel Newton method. The results do not provide much insight, however, into the performance gains compared with the sequential Newton method. To analyze this case, we interpret the parallel Newton method as a manifold-tracking algorithm [20]. In particular, the method can be seen as a *warm-starting* technique that stays close to a moving manifold formed by the upcoming data x_k^j from the neighboring time step. In other words, the method stays close to the manifold given by the solution of the system

$$\Phi(x_{k+1}, x_k^j) = 0,\ j = 0, ..., J_k, \quad (38)$$

which is denoted as $x_{k+1}^*(x_k^j)$. We will see that each iteration of the parallel Newton method x_{k+1}^j stays close and eventually converges to the manifold solution. Therefore, once x_k^j converges to x_k^*, a few (usually one) extra Newton iterations will be needed to converge to $x_{k+1}^*(x_k^*)$. This provides the savings compared with the sequential method, which converges the system $\Phi(x_{k+1}, x_k^*) = 0$ from the initial guess x_{k+1}^0 once x_k^* is known. These savings can be significant. For instance, if the sequential method takes J_{seq} iterations per time step k and the parallel Newton method takes one iteration per step, then the total iteration savings are $(J_{seq} - 1) \cdot N$. As can be seen, the savings scale with the length of the time horizon.

The following theorem establishes conditions under which the parallel Newton method converges to the moving manifold $x^*_{k+1}(x^j_k)$. To prove this, we compare the distances $\|x^{j+1}_{k+1}(p_j) - x^*_{k+1}(p_j)\|$ and $\|x^j_{k+1}(p_{j-1}) - x^*_{k+1}(p_{j-1})\|$, where $p_j := x^j_k$ and $p_{j-1} := x^{j-1}_k$. For simplicity, we will drop the subindex $k+1$ and express x_{k+1} simply as x. Accordingly, the Newton system can be written as

$$\begin{aligned}&\Phi(x^j(p_{j-1}), p_j)\\ &\qquad + \Phi'(x^j(p_{j-1}), p_j)(x^{j+1}(p_j) - x^j(p_{j-1})) = 0 \end{aligned} \tag{39}$$

Theorem 3: (**Convergence to Moving Manifold**). Assume that the Jacobian $\Phi'(x^j(p_{j-1}), p_j)$ is nonsingular. If $(\|x^*(p_j) - x^*(p_{j-1})\| + \|x^j(p_{j-1}) - x^*(p_{j-1})\|)^2 = O(\|x^j(p_{j-1}) - x^*(p_{j-1})\|)$, then the iteration sequence $x^{j+1}(p_j)$ given by (39) converges to the manifold $x^*(p_j)$ superlinearly. If $\|x^*(p_j) - x^*(p_{j-1})\| = O(\|x^j(p_{j-1}) - x^*(p_{j-1})\|)$ then convergence is quadratic.

Proof: We have

$$\begin{aligned}0 &= \Phi(x^*(p_j), p_j)\\ &= \Phi(x^*(p_{j-1}), p_j)\\ &\quad + \int_0^1 \Phi'(x^*(p_{j-1}) + \tau(x^*(p_j) - x^*(p_{j-1})), p_j)\cdot\\ &\qquad \cdot (x^*(p_j) - x^*(p_{j-1}))d\tau \end{aligned} \tag{40}$$

and

$$\begin{aligned}\Phi(x^*(p_{j-1}), p_j) &= \Phi(x^j(p_{j-1}), p_j)\\ &+ \int_0^1 \Phi'(x^j(p_{j-1}) + \tau(x^*(p_{j-1}) - x^j(p_{j-1})), p_j)\cdot\\ &\qquad \cdot (x^*(p_{j-1}) - x^j(p_{j-1}))d\tau. \end{aligned} \tag{41}$$

Combining (41) and (39), we have

$$\begin{aligned}\Phi(x^*(p_{j-1}), p_j) &= -\Phi'(x^j, p_j)(x^{j+1}(p_j) - x^j(p_{j-1}))\\ &+ \int_0^1 \Phi'(x^j(p_{j-1}) + \tau(x^*(p_j) - x^j(p_{j-1})), p_j)\cdot\\ &\quad \cdot (x^*(p_j) - x^j(p_{j-1}))d\tau. \end{aligned} \tag{42}$$

Combining (40) and (42), we have

$$\begin{aligned}&\Phi'(x^j(p_{j-1}), p_j)(x^{j+1}(p_j) - x^j(p_{j-1}))\\ &= \int_0^1 \Phi'(x^*(p_{j-1}) + \tau(x^*(p_j) - x^*(p_{j-1})), p_j)\cdot\\ &\quad \cdot (x^*(p_j) - x^*(p_{j-1}))d\tau\\ &\quad + \int_0^1 \Phi'(x^j(p_{j-1}) + \tau(x^*(p_{j-1}) - x^j(p_{j-1})), p_j)\cdot\\ &\quad \cdot (x^*(p_j) - x^j(p_{j-1}))d\tau. \end{aligned} \tag{43}$$

or,

$$\begin{aligned}&\Phi'(x^j(p_{j-1}), p_j)\cdot\\ &\cdot (x^{j+1}(p_j) - x^*(p_j) + x^*(p_j)\\ &\qquad - x^*(p_{j-1}) + x^*(p_{j-1}) - x^j(p_{j-1}))\\ &= \int_0^1 \Phi'(x^*(p_{j-1}) + \tau(x^*(p_j) - x^*(p_{j-1})), p_j)\cdot\\ &\quad \cdot (x^*(p_j) - x^*(p_{j-1}))d\tau\\ &\quad + \int_0^1 \Phi'(x^j(p_{j-1}) + \tau(x^*(p_{j-1}) - x^j(p_{j-1})), p_j)\cdot\\ &\quad \cdot (x^*(p_{j-1}) - x^j(p_{j-1}))d\tau. \end{aligned}$$

Rearranging and bounding terms, we have

$$\begin{aligned}&\|\Phi'(x^j(p_{j-1}), p_j)\|\|x^{j+1}(p_j) - x^*(p_j)\|\\ &= O(\|x^j(p_{j-1}) - x^*(p_{j-1})\| + \|x^*(p_j) - x^*(p_{j-1})\|)^2. \end{aligned} \tag{44}$$

Under the assumptions for $\|x^*(p_j) - x^*(p_{j-1})\|$ the result follows. □

Corollary 1: Assume conditions of Theorem 1 hold. Then, there exist iterates J_k such that $p_{J_k} = p_{J_k - 1} = x^*_k$ for all $k = 1, ..., N$ and $\kappa \geq 0$ such that

$$\begin{aligned}&\|x^{J_k+1}_{k+1}(x_{J_k}) - x^*_{k+1}(x^*_k)\|\\ &\qquad \leq \kappa \|x^{J_k}_{k+1}(x^{J_k-1}_k) - x^*_{k+1}(x^{J_k-1}_k)\|^2, \end{aligned} \tag{45}$$

for $k = 0, ..., N-1$.

Proof: Since there exists $j = J_1$ at which $\|x^j_1 - x^{j-1}_1\| = 0$ then $\|x^*_2(x^j_1) - x^*_2(x^{j-1}_1)\| = 0$ and quadratic convergence is attained at time step $k = 2$ at iteration $J_1 + 1$. The result follows by applying a recursive argument. □

As can be seen, the parallel Newton scheme will converge quadratically to the solution $x^*_{k+1}(x^*_k)$ once it is positioned at the moving manifold $x^*_{k+1}(x^j_k)$. In Figure 1 we illustrate this behavior.

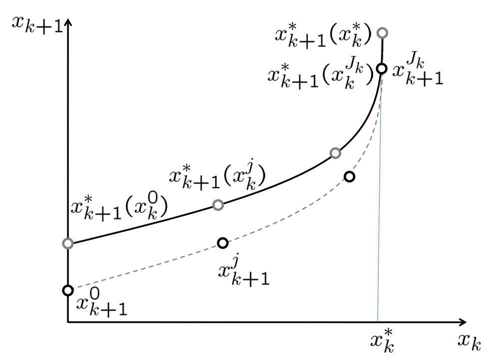


Fig. 1: Convergence of Parallel Newton Method to Moving Manifold.

V. Power System Model

Figure 2 shows the circuit representation of the two-axis synchronous machine. This model was chosen since it accounts for transient responses and neglects the effects of subtransients [21], [22].

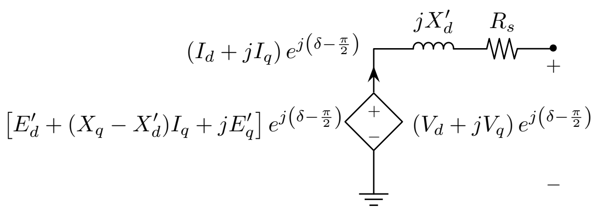


Fig. 2: Two-Axis Machine Model

Consider an n bus network with m machines attached to it. The differential equations for each machine $i = 1, 2, \ldots, m$ are given by

$$\begin{aligned} T'_{doi}\frac{dE'_{qi}}{dt} &= -E'_{qi} - (X_{di} - X'_{di})I_{di} + E_{fdi} && (46)\\ T'_{qoi}\frac{dE'_{di}}{dt} &= -E'_{di} + (X_{qi} - X'_{qi})I_{qi} && (47)\\ \frac{d\delta_i}{dt} &= \omega_i - \omega_s && (48)\\ \frac{2H_i}{\omega_s}\frac{d\omega_i}{dt} &= T_{Mi} - E'_{di}I_{di} - E'_{qi}I_{qi} \\ &\quad -(X'_{qi} - X'_{di})I_{di}I_{qi} - T_{FWi}, && (49) \end{aligned}$$

where Park's transformation has been used to switch the coordinate system. The exciter for the machines is an IEEE type 1 exciter given by

$$\begin{aligned} T_{Ei}\frac{dE_{fdi}}{dt} &= -(K_{Ei} + S_{Ei}(E_{fdi}))E_{fdi} + V_{Ri} && (50)\\ T_{Fi}\frac{dR_{fi}}{dt} &= -R_{fi} + \frac{K_{Fi}}{T_{Fi}}E_{fdi} && (51)\\ T_{Ai}\frac{V_{Ri}}{dt} &= -V_{Ri} + K_{Ai}R_{Fi} - \frac{K_{Ai}K_{Fi}}{T_{Fi}}E_{fdi} \\ &\quad +K_{Ai}(V_i^{ref} - |V_i|), && (52) \end{aligned}$$

with $|V_i| = \sqrt{V_{di}^2 + V_{qi}^2}$. For the remainder of this paper, it is assumed that every machine is driven by a steam turbine that is modeled as

$$\begin{aligned} T_{CHi}\frac{dT_{Mi}}{dt} &= -T_{Mi} + P_{SVi} && (53)\\ T_{SVi}\frac{dP_{SVi}}{dt} &= -P_{SVi} + P_{Ci} - \frac{1}{R_{Di}}\left(\frac{\omega_i}{\omega_s} - 1\right), && (54) \end{aligned}$$

where (53) is the behavior of a non-reheat steam turbine model and (54) is the speed governor for the system.

The power balance equations of the network at each bus are given by

$$\begin{aligned} 0 &= \sum_{k=1}^{n}[V_iV_kY_{ik}^*e^{j(\theta_i-\theta_k)} - (P_i + jQ_i)] \\ &\quad -V_ie^{j\theta_i}(I_{di} - jI_{qi})e^{-j(\delta_i-\pi/2)} && (55)\\ 0 &= \sum_{k=1}^{n}[V_jV_kY_{jk}^*e^{j(\theta_j-\theta_k)} - (P_j + jQ_j)], && (56) \end{aligned}$$

where (55) includes the complex power delivered to the generation buses $i = 1, 2, \ldots, m$ and (56) represents the load buses $j = m + 1, \ldots, n$.

From Fig. 2, the coupling between the dynamic states and the network equations is

$$\begin{bmatrix} R_{si} & -X'_{qi} \\ X'_{di} & R_{si} \end{bmatrix} \begin{bmatrix} I_{di} \\ I_{qi} \end{bmatrix} = \begin{bmatrix} E'_{di} - V_i\sin(\delta_i - \theta_i) \\ E'_{qi} - V_i\cos(\delta_i - \theta_i) \end{bmatrix}, \quad (57)$$

where the terminal voltage $V_ie^{j\theta_i}$ perceived by the system bus i is $(V_d + jV_q)e^{j(\delta_i - \frac{\pi}{2})}$.

Simplifying (46)-(57) one can write the power system as a differential algebraic equation of the form (2). The dynamic state variables are given by $x_i = [E'_{qi}, E'_{di}, \delta_i, \omega_i, E_{fdi}, R_{fi}, V_{Ri}, T_{Mi}, P_{SVi}]^T$ $\forall i = 1, 2, \ldots, m$. The algebraic variables are $y_i = [I_{di}, I_{qi}, V_i, \theta_i]^T$ $\forall i = 1, 2, \ldots, m$ and $y_j = [V_j, \theta_j]^T$ $\forall j = m + 1, \ldots, n$. To solve the differential model, we apply a trapezoidal discretization rule. For an m machine, n bus system, the number of states that need to be solved is $11m + 2n$. An entire interconnect can easily contain $m = 1,000$ machines and $n = 10,000$ nodes giving a system of $31,000$ DAEs.

VI. Case Studies

A case study was performed on the Western Electricity Coordinating Council's (WECC) 3 machine, 9 bus test system shown in Fig. 3 [21]. At $t = 0.04$ sec a fault is created by severing the transmission lines that connect buses $\{4, 5\}$ and $\{8, 9\}$. The fault is cleared after 0.15 sec, and the transient response is computed with the traditional sequential approach and in parallel. The simulation has a duration of 10 sec and uses a step size of $h = 0.01$ sec for the numerical integration. The simulations are currently implemented in MATLAB, and the number of iterations required for the global residual to converge less than 10^{-10} for each solution is recorded. The number of iterations is used as the metric to compare results of the two methods since the computational time cannot realistically determined until the systems are tested in a parallel environment.

Figures 4(a) and 4(b) are the frequency responses of the system using the sequential and parallel methods, respectively. In addition to the frequency response, the same results were achieved for the remaining 50 states.

Figure 5(a) shows the local residuals of the first 80 iterations for the sequential method. On average, 4 iterations were needed to solve the set of nonlinear equations at each time step. Notice the two prominent spikes in the number of iterations at times

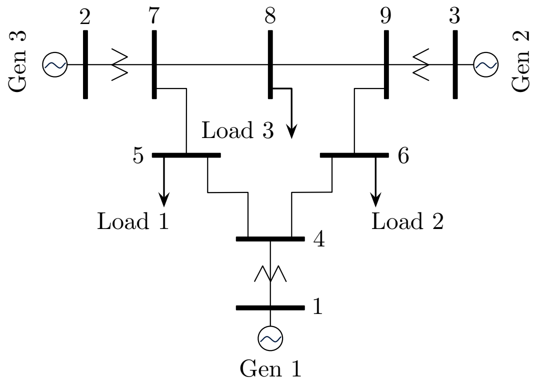
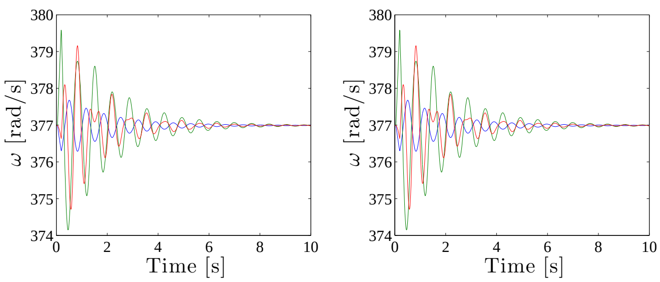


Fig. 3: WECC 3 Machine 9 Bus Network.



(a) Sequential Method

(b) Parallel Method

Fig. 4: Network Frequency Response.

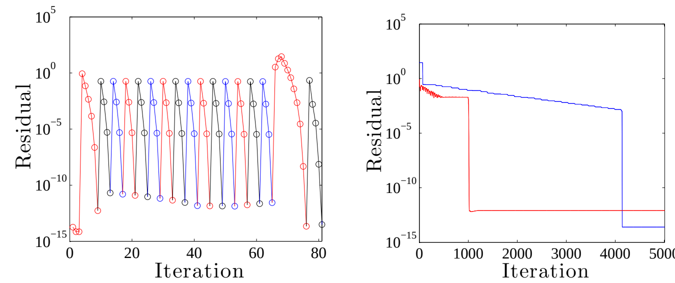
4 and 66 resulting from the introduction and removal of the fault, respectively.

The global residuals are compared in Fig. 5(b). The sequential method required *4,142 iterations* to converge whereas the parallel method converged within the desired tolerances after *1,017 iterations* for the 1,000-time-step simulation, which is close to the theoretical lower bound. This desirable performance is obtained despite the large disturbance introduced by the fault. This increased robustness of the parallel method is attributed to the close tracking of the moving solution manifold. The parallel method resulted in a 75% savings in the number of iterations over the sequential method. Additionally, we observed that as the length of the horizon is increased, the average number of iterations of the parallel method asymptotically converges to the theoretical lower bound.

VII. Conclusions and Future Work

We provide a convergence analysis that explains why parallel block Newton methods can significantly reduce the number of iterations compared to sequential methods. In addition, we provide a theoretical lower bound on the number of iterations needed by the method. Numerical tests indicate that the method is scalable.

Future work will take advantage of the linear algebra structure of the problem to develop a multilevel parallelization scheme to reduce the time of each parallel Newton iteration. This will also enable us to group multiple time steps in a single Newton system and thus achieve convergence properties that are closer to full-space Newton methods. Additionally, strategies with adaptive time steps will be implemented. Moreover, an important research direction is the convergence of more general multi-splitting (e.g., parallel in space) methods.



(a) Local Residual of Sequential Method

(b) Global Residual Comparison

Fig. 5: Residuals.

Acknowledgments

This work was supported by the U.S. Department of Energy, under Contract No. DE-AC02-06CH11357.

References

[1] U.S.-Canada Power System Outage Task Force, “Final report on the august 14, 2003 blackout in the United States and Canada: Causes and recommendations,” *https://reports.energy.gov/BlackoutFinal-Web.pdf*, 2004.

[2] F. L. Alvarado, “Parallel solution of transient problems by trapezoidal integration,” *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-98, pp. 1080 –1090, May 1979.

[3] J. S. Chai and A. Bose, “Bottlenecks in parallel algorithms for power system stability analysis,” *IEEE Transactions on Power Systems*, vol. 8, no. 1, pp. 9 – 15, February 1993.

[4] K. W. Chan, R. W. Dunn, and A. R. Daniels, “Efficient heuristic partitioning algorithm for parallel processing of large power systems network equations,” *IEE Proceedings on Generation, Transmission, and Distribution*, vol. 142, no. 6, pp. 625 – 630, November 1995.

[5] M. Ilic’-Spong, M. L. Crow, and M. A. Pai, “Transient stability simulation by waveform relaxation methods,” *IEEE Transactions on Power Systems*, vol. 2, no. 4, pp. 943 – 949, November 1987.

[6] M. L. Crow and M. Ilic, “The parallel implementation of the waveform relaxation method for transient stability simulations,” *IEEE Transactions on Power Systems*, vol. 5, no. 3, pp. 922 – 932, August 1990.

[7] I. C. Decker, D. M. Falcao, and E. Kaszkurewicz, “Parallel implementation of a power system dynamic simulation methodology using the conjugate gradient method,” in *Power Industry Computer Application Conference*, May 1991, pp. 245 – 252.

[8] M. A. Pai, P. W. Sauer, and A. Y. Kulkarni, “Conjugate gradient approach to parallel processing in dynamic simulation of power systems,” in *American Controls Conference*, June 1992, pp. 1644 – 1647.

[9] I. C. Decker, D. M. Falcao, and E. Kaszkurewicz, “Conjugate gradient methods for power system dynamic simulation on parallel computers,” *IEEE Transactions on Power Systems*, vol. 11, no. 3, pp. 1218 – 1227, August 1996.

[10] A. Y. Kulkarni, M. A. Pai, and P. W. Sauer, “Iterative solver techniques in fast dynamic calculations of power systems,” *International Journal of Electrical Power & Energy Systems*, vol. 23, no. 3, pp. 237 – 244, March 2001.

[11] M. L. Scala, A. Bose, and D. J. Tylavsky, “A relaxation type multigrid parallel algorithm for power system transient stability analysis,” in *IEEE International Symposium on Circuits and Systems*, vol. 3, May 1989, pp. 1954 –1957.

[12] M. L. Scala, A. Bose, D. J. Tylavsky, and J. S. Chai, "A highly parallel method for transient stability analysis," *IEEE Transactions on Power Systems*, vol. 5, no. 4, pp. 1439 –1446, November 1990.

[13] J. S. Chai, N. Zhu, A. Bose, and D. J. Tylavsky, "Parallel Newton type methods for power system stability using local and shared memory multiprocessors," *IEEE Transactions on Power Systems*, vol. 6, no. 4, pp. 1539 – 1545, November 1991.

[14] J. Q. Wu, A. Bose, J. A. Huang, A. Valette, and F. Lafrance, "Parallel implementation of power system transient stability analysis," *IEEE Transactions on Power Systems*, vol. 10, no. 3, pp. 1226 – 1233, August 1995.

[15] S. G. Nash and A. Sofer, "Block truncated-Newton methods for parallel optimization," *Mathematical Programming*, vol. 45, pp. 529–546, 1989, 10.1007/BF01589117. [Online]. Available: http://dx.doi.org/10.1007/BF01589117

[16] A. I. Zecevik and D. D. Siljak, "A block-parallel newton method via overlapping epsilon decompositions," vol. 15, no. 3, pp. 824–844, 1994. [Online]. Available: http://dx.doi.org/doi/10.1137/S0895479892229115

[17] M. La Scala, M. Brucoli, F. Torelli, and M. Trovato, "A gauss-jacobi-block-newton method for parallel transient stability analysis [of power systems]," *Power Systems, IEEE Transactions on*, vol. 5, no. 4, pp. 1168 –1177, nov 1990.

[18] U. M. Ascher and L. R. Petzold, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Philadelphia, PA: SIAM, 1998.

[19] S. M. Robinson, "Strongly regular generalized equations," *Mathematics of Operations Research*, vol. 5, pp. 43–61, 1980.

[20] V. M. Zavala and M. Anitescu, "Real-time nonlinear optimization as a generalized equation," vol. 48, no. 8, pp. 5444–5467, 2010. [Online]. Available: http://dx.doi.org/doi/10.1137/090762634

[21] P. W. Sauer and M. A. Pai, *Power System Dynamics and Stability*. Champaign, IL: Stipes Publishing L.L.C., 2006.

[22] P. M. Anderson and A. A. Fouad, *Power System Control and Stability*, 2nd ed. Hoboken, NJ: Wiley-IEEE Press, 2003.